



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/672,180	09/26/2003	Jeffrey G. Cheng	00100.03.0032	9865

29153 7590 06/02/2006

ATI TECHNOLOGIES, INC.
C/O VEDDER PRICE KAUFMAN & KAMMHOLZ, P.C.
222 N.LASALLE STREET
CHICAGO, IL 60601

EXAMINER

URICK, MATTHEW T

ART UNIT	PAPER NUMBER
----------	--------------

2113

DATE MAILED: 06/02/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/672,180	Applicant(s) CHENG ET AL.	
	Examiner Matt Urick	Art Unit 2113	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 26 September 2003.
- 2a) ☐ This action is FINAL. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-25 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 22 and 23 is/are allowed.
- 6) ☒ Claim(s) 1-21, 24 and 25 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on 26 September 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

Non-Final Official Action

Status of the Claims

Claims 1-3, 5-8, 10, 11, 14, 15, 17, 18, 24, and 25 are rejected under 35 USC 102

Claims 4, 9, 12, 13, 16, 19-21 are rejected under 35 USC 103

Claims 22 and 23 are allowable

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1-3, 5-8, 10, 11, 14, 15, 17, 18, 24, and 25 are rejected under 35

U.S.C. 102(e) as being anticipated by Forsman (United States Patent No. 6,742,139).

As per claim 1, Forsman discloses:

A circuit for monitoring and resetting a co-processor comprising:

a hang detector module operative to detect a hang in the co-processor (column 4 lines 25-27); and

a selective processor reset module operative to selectively reset the co-processor without resetting a processor, in response to detecting a hang in the co-

Art Unit: 2113

processor (column 4 lines 27-35).

As per claim 2, Forsman discloses:

The system of claim 1 wherein an operating system executes on the processor (column 4 lines 4-5).

As per claim 3, Forsman discloses:

The system of claim 1 wherein the hang detector module detects the hang in the co-processor by detecting a discrepancy between a current state of the co-processor and a current activity of the co-processor (column 4 lines 9-12, lines 25-35).

As per claim 5, Forsman discloses:

The system of claim 1 further comprising:

a halt communications module operative to halt command communications with the co-processor, in response to detecting a hang in the co-processor (column 5 lines 10-15);

a reset check module operative to detect if the co-processor has been successfully reset, in response to the resetting of the co-processor (column 5 lines 28-33); and

a restart communications module operative to restart command communications with the co-processor, in response to detecting that the co-processor has been

successfully reset (column 5 lines 33-35).

As per claim 6, Forsman discloses:

A method of monitoring and resetting a co-processor comprising the steps of:
detecting a hang in the co-processor (column 4 lines 25-27); and
selectively resetting the co-processor without resetting a processor, in response
to detecting a hang in the co-processor (column 4 lines 27-35).

As per claim 7, Forsman discloses:

The method of claim 6 wherein an operating system executes on the processor
(column 4 lines 4-5).

As per claim 8, Forsman discloses:

The method of claim 6 wherein detecting a hang in the co-processor is performed
by detecting a discrepancy between a current state of the co-processor and a current
activity of the co-processor (column 4 lines 9-12, lines 25-35).

As per claim 10, Forsman discloses:

The method of claim 6 further comprising the step of:
halting command communications with the co-processor, in response to
detecting a hang in the co-processor (column 5 lines 10-15);

Art Unit: 2113

detecting if the co-processor has been successfully reset, in response to the resetting of the co-processor(column 5 lines 28-33); and

restarting command communications with the co-processor, in response to detecting that the co-processor has been successfully reset (column 5 lines 33-35).

As per claim 11, Forsman discloses:

A circuit for monitoring and resetting a co-processor comprising:

a hang detector module operative to detect a hang in the co-processor (column 4 lines 25-27);

a halt communications module operative to halt command communications with the co-processor, in response to detecting a hang in the co-processor (column 5 lines 10-15);

a selective processor reset module operative to selectively reset the co-processor without resetting a processor, in response to detecting a hang in the co-processor (column 4 lines 27-35);

a reset check module operative to detect if the co-processor has been successfully reset, in response to the resetting of the co-processor (column 5 lines 28-33); and

a restart communications module operative to restart command communications with the co-processor, in response to detecting that the co-processor has been successfully reset (column 5 lines 33-35).

Art Unit: 2113

As per claim 14, Forsman discloses:

A system for monitoring and resetting a co-processor comprising:

a processor and a co-processor (column 4 lines 10-12); and

a memory containing instructions including:

hang detector module instructions operative to detect a hang in the co-processor (column 4 lines 25-27); and

selective processor reset module instructions operative to selectively reset the co-processor without resetting the processor, in response to detecting a hang in the co-processor (column 4 lines 27-35).

As per claim 15, Forsman discloses:

The circuit of claim 14 wherein the processor is a host processor and the memory contains operating system instructions (column 3 lines 14-20).

As per claim 17, Forsman discloses:

The system of claim 14 further comprising: reset check module instructions operative to detect if the co-processor has been successfully reset, in response to the resetting of the co-processor (column 5 lines 28-33).

As per claim 18, Forsman discloses:

The system of claim 14 further comprising:

halt communications module instructions operative to halt command communications with the co-processor, in response to detecting a hang in the co-processor (column 5 lines 10-15); and

restart communications module instructions operative to restart command communications with the co-processor, in response to detecting that the co-processor has been successfully reset (column 5 lines 33-35).

As per claim 24, Forsman discloses:

A memory containing instructions executable on a processor that causes the processor to:

detect a hang in a co-processor (column 4 lines 25-27); and

selectively reset the co-processor without resetting the processor, in response to detecting a hang in the co-processor (column 4 lines 27-35).

As per claim 25, Forsman discloses:

The memory of claim 24 further including instructions that causes the processor to:

detect if the co-processor has been successfully reset, in response to the resetting of the co-processor (column 5 lines 28-33).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 4, 9, and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Forsman (United States Patent No. 6,742,139) in view of Kahle (United States Patent No. 6,543,002).

As per claim 4, Forsman discloses:

The system of claim 3 wherein the discrepancy is detected by detecting the current state to be busy, by detecting a busy flag to be set (column 4 lines 9-12),

Forsman fails to disclose:

and detecting no progress on current activity, by detecting the same contents in a co-processor register as examined before and after a wait period.

Kahle discloses a system in which a processor asserts a "completion" signal each time an instruction is completed (column 6 lines 13-24). If the completion signal is not asserted over a certain timeout period, the processor issues a "stop" signal and subsequently a "flush" signal (column 6 line 52 - column 7 line 6), flushing the instruction registers (column 1 lines 62-66). Kahle discloses that this system enables a processor to recover from a "hang" condition without rebooting the entire system (column 1 lines 31-40). Forsman also discloses that he wishes to reset a processor

Art Unit: 2113

without resetting the entire computing system, as this would be a burden on the user (column 4 lines 12-24). Forsman discloses that a heartbeat or other indication may be used to determine if the processor is acting correctly (column 5 lines 28-35). Kahle's monitoring system would enable constant monitoring of the processor with quick and seamless recovery. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the monitoring system of Kahle into the processor reset system of Forsman, providing a reliable indication of processor activity and a seamless recovery.

As per claim 9, Forsman discloses:

The method of claim 8 wherein the wherein the discrepancy is detected by detecting the current state to be busy, by detecting a busy flag to be set set (column 4 lines 9-12)

Forsman does not disclose:

detecting no progress on current activity, by detecting the same contents in a co-processor register as examined before and after a wait period.

Kahle discloses a system in which a processor asserts a "completion" signal each time an instruction is completed (column 6 lines 13-24). If the completion signal is not asserted over a certain timeout period, the processor issues a "stop" signal and subsequently a "flush" signal (column 6 line 52 - column 7 line 6), flushing the instruction registers (column 1 lines 62-66). Kahle discloses that this system enables a

Art Unit: 2113

processor to recover from a "hang" condition without rebooting the entire system (column 1 lines 31-40). Forsman also discloses that he wishes to reset a processor without resetting the entire computing system, as this would be a burden on the user (column 4 lines 12-24). Forsman discloses that a heartbeat or other indication may be used to determine if the processor is acting correctly (column 5 lines 28-35). Kahle's monitoring system would enable constant monitoring of the processor with quick and seamless recovery. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the monitoring system of Kahle into the processor reset system of Forsman, providing a reliable indication of processor activity and a seamless recovery.

As per claim 16, Forsman discloses:

The system of claim 14 wherein the hang detector module instructions detect the hang in the co-processor by detecting the current state to be busy, as reflected in a busy flag (column 4 lines 9-12)

Forsman does not disclose:

detecting no progress on current activity, as reflected the absence of co-processor register activity.

Kahle discloses a system in which a processor asserts a "completion" signal each time an instruction is completed (column 6 lines 13-24). If the completion signal is not asserted over a certain timeout period, the processor issues a "stop" signal and

Art Unit: 2113

subsequently a “flush” signal (column 6 line 52 - column 7 line 6), flushing the instruction registers (column 1 lines 62-66). Kahle discloses that this system enables a processor to recover from a “hang” condition without rebooting the entire system (column 1 lines 31-40). Forsman also discloses that he wishes to reset a processor without resetting the entire computing system, as this would be a burden on the user (column 4 lines 12-24). Forsman discloses that a heartbeat or other indication may be used to determine if the processor is acting correctly (column 5 lines 28-35). Kahle’s monitoring system would enable constant monitoring of the processor with quick and seamless recovery. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the monitoring system of Kahle into the processor reset system of Forsman, providing a reliable indication of processor activity and a seamless recovery.

Claims 12, 13, and 19-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Forsman (United States Patent No. 6,742,139) in view of Hill (United States Patent Application Publication No. 2002/0093505).

As per claim 12, Forsman does not disclose:

The circuit of claim 11 wherein the processor is a host processor and the co-processor is a graphics processor.

Hill discloses a system in which a graphics processor works with the main CPU to execute graphic-intensive software (§ 21 – graphics processor referred to as “graphic accelerator”). Hill’s system prevents the entire computing system from crashing due to a

Art Unit: 2113

failure specific to the graphics accelerator, by performing a series of tests (§ 23 lines 1-16). If the graphics accelerator does not perform adequately, the software graphics processor will take over instead of crashing the system (§ 34 last 4 lines, § 35). Hill discloses that he intends to improve situations where a reset due to a graphics processor crash is unacceptable (§ 19 last 8 lines). Forsman also wishes to recover a system of multiple processors without performing a full system reset (Forsman column 1 lines 25-29). Using a graphics processor in place of a service processor in Forsman's system would prevent system crashes due to graphics accelerators as well as the service processor, preventing crashes due to incompatible video accelerators or other faults. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the graphics accelerator monitoring system of Hill into the processor reset system of Forsman, providing additional protection against hardware failure.

As per claim 13, Forsman discloses:

The circuit of claim 12 wherein the hang detector module detects the hang in the graphics processor by detecting a discrepancy between a current state of the graphics processor and a current activity of the graphics processor (column 4 lines 9-12, lines 25-35).

As per claim 19, Forsman discloses:

A system for monitoring and resetting a co-processor comprising:

a host processor (column 4 lines 10-12);
a memory containing instructions including:
hang detector module instructions operative to detect a hang in the [graphics]
processor (column 4 lines 25-27); and
selective processor reset module instructions operative to selectively reset the
[graphics] processor without resetting the host processor, in response to detecting a
hang in the graphics processor (column 4 lines 27-35).

Forsman does not disclose:

a graphics processor;

Hill discloses a system in which a graphics processor works with the main CPU to execute graphic-intensive software (§ 21 – graphics processor referred to as “graphic accelerator”). Hill’s system prevents the entire computing system from crashing due to a failure specific to the graphics accelerator, by performing a series of tests (§ 23 lines 1-16). If the graphics accelerator does not perform adequately, the software graphics processor will take over instead of crashing the system (§ 34 last 4 lines, § 35). Hill discloses that he intends to improve situations where a reset due to a graphics processor crash is unacceptable (§ 19 last 8 lines). Forsman also wishes to recover a system of multiple processors without performing a full system reset (Forsman column 1 lines 25-29). Using a graphics processor in place of a service processor in Forsman’s system would prevent system crashes due to graphics accelerators as well as the service processor, preventing crashes due to incompatible video accelerators or other

Art Unit: 2113

faults. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the graphics accelerator monitoring system of Hill into the processor reset system of Forsman, providing additional protection against hardware failure.

As per claim 20, Forsman discloses:

The system of claim 19 wherein the hang detector module instructions detect the hang in the graphics processor by detecting the current state of the graphics processor to be busy, as reflected in a graphics processor busy flag, and detecting the current activity of the graphics processor to be idle, as reflected in the absence of graphics processor register activity (column 4 lines 9-12, lines 25-35).

As per claim 21, Forsman discloses:

The system of claim 19 further comprising:

halt communications module instructions operative to halt rendering command communications with the graphics processor, in response to detecting a hang in the graphics processor (column 5 lines 10-15);

reset check module instructions operative to detect if the graphics processor has been successfully reset, in response to the resetting of the graphics processor (column 5 lines 28-33); and

restart communications module instructions operative to restart rendering command communications with the graphics processor, in response to detecting that the graphics processor has been successfully reset (column 5 lines 33-35).

Hill discloses a system in which a graphics processor works with the main CPU to execute graphic-intensive software (§ 21 – graphics processor referred to as “graphic accelerator”). Hill’s system prevents the entire computing system from crashing due to a failure specific to the graphics accelerator, by performing a series of tests (§ 23 lines 1-16). If the graphics accelerator does not perform adequately, the software graphics processor will take over instead of crashing the system (§ 34 last 4 lines, § 35). Hill discloses that he intends to improve situations where a reset due to a graphics processor crash is unacceptable (§ 19 last 8 lines). Forsman also wishes to recover a system of multiple processors without performing a full system reset (Forsman column 1 lines 25-29). Using a graphics processor in place of a service processor in Forsman’s system would prevent system crashes due to graphics accelerators as well as the service processor, preventing crashes due to incompatible video accelerators or other faults. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the graphics accelerator monitoring system of Hill into the processor reset system of Forsman, providing additional protection against hardware failure.

Allowable

Claim 22 states:

A system for monitoring and resetting a co-processor comprising:

a host processor;

a graphics processor; and

a memory containing instructions including:

hang detector module instructions operative to detect a hang in the graphics processor by detecting the current state to be busy, by detecting a busy flag to be set, and detecting no progress on current activity, by detecting the same contents in a co-processor register as examined before and after a wait period;

halt communications module instructions operative to halt rendering command communications with the graphics processor by setting a send flag to an off state and setting a receive flag to an off state, in response to detecting a hang in the graphics processor;

save snapshot module instructions operative to save a snapshot of the hardware and software status including any one or more of the following: graphics register data, graphics command queue data, chipset info, and AGP bus status, in response to the detecting a hang in the graphics processor;

selective processor reset module instructions operative to selectively reset the graphics processor without resetting the host processor, in response to detecting a hang in the graphics processor;

reset check module instructions operative to detect if the graphics processor has been successfully reset, in response to the resetting of the graphics processor;

display mode switch module instructions operative to perform a display mode switch, in response to the selectively resetting of the graphics processor;

functioning check module instructions operative to detect if the graphics processor is fully functioning, in response to the resetting of the graphics processor;

restart communications module instructions operative to restart rendering command communications with the graphics processor by setting the send flag to an on state, in response to detecting that the graphics processor has been successfully reset; and

software rendering module instructions operative to dynamically switch to software rendering mode, in response to detecting any one or more of the following: a detection that the graphics processor has not been successfully reset and a detection that the graphics processor is not fully functioning.

Claim 23 states:

The system of claim 22 further comprising:

hang resolved prompt module instructions operative to display a prompt indicating that a hang was detected and resolved, in response to detecting that the graphics processor is fully functioning;

report send prompt module instructions operative to display a prompt requesting an input as to whether a user wants to have an error report sent to a remote location, in response to detecting a hang in the graphics processor;

report send module instructions operative to send an error report to a remote location including the hardware and software status, in response to receiving a user request to send an error report to a remote location; and

hang unresolved prompt module instructions operative to display a prompt indicating that a hang was detected and cannot be resolved without a reset of the host processor being performed, in response to detecting one or more of the following: the graphics processor was not successfully reset and the graphics processor is not fully functioning.

Claims 22 and 23 were examined and considered allowable over the prior art. As disclosed by the above mentioned references from Forsman, Hill, and Kahle, systems exist which perform error recovery in a second processor, such as a graphics processor or a service processor, without resetting the main CPU. These systems are also able to ensure that the second processor was reset correctly before resuming operation. However, of the above mentioned references, teach the method of producing a save snapshot module able to store graphics register data, graphics command queue data, chipset info, and AGP bus status, in response to the detecting a hang in the graphics processor; as well as software rendering module instructions operative to dynamically

Art Unit: 2113

switch to software rendering mode, in response to detecting a detection that the graphics processor has not been successfully reset and a detection that the graphics processor is not fully functioning. The above mentioned references do not teach these limitations, nor would it have been obvious to combine them with the other limitations of claim 22.

Claim 23 is considered allowable as being dependant on claim 22.

Conclusion

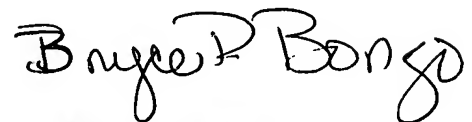
Any inquiry concerning this communication or earlier communications from the examiner should be directed to Matt Urick whose telephone number is (571) 272-0805. The examiner can normally be reached on 8:00 - 4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert Beausoliel can be reached on (571) 272-3645. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2113

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

MTZ


BRYCE P. BONZO
PRIMARY EXAMINER